A GRAPHICAL USER INTERFACE HAVING AN ATTACHED TOOLBAR FOR DRAG AND DROP EDITING IN DETAIL-IN-CONTEXT LENS PRESENTATIONS

This application claims priority from Canadian Patent Application No. 2,393,887 filed July 17, 2002, the disclosure of which is incorporated herein by reference.

FIELD OF THE INVENTION

This invention relates to the field of computer graphics processing, and more specifically, to a method and system for performing accurate drag and drop ("DAD") operations on digital images using detail-in-context lenses and a detail-in-context graphical user interface ("GUI") having an attached toolbar.

BACKGROUND OF THE INVENTION

Most modern computer software employs a graphical user interface ("GUI") to convey information to and receive commands from users. A graphical user interface relies on a variety of GUI objects or controls, including icons, toolbars, drop-down menus, text, dialog boxes, buttons, and the like. In such GUI systems, toolbars provide an effective way to display numerous computer commands or controls. Toolbars usually include buttons, which are arranged in one or more rows or columns. Each button is associated with a command, and is identified by an icon that represents or depicts that command. For example, the "print" command may be invoked by clicking on a button whose icon depicts a printer. Advantageously, a user can invoke commands on the toolbar by clicking once on the associated button. In addition to buttons, toolbars can also include other interactive controls, such as text boxes, combo boxes, etc. Some toolbars can be turned on or off, and can be oriented horizontally or vertically. Although most toolbars are visually attached to a window, some may float above a window. In some programs that employ toolbars, the toolbars can be modified by adding or deleting controls, or by changing the function associated with a control. This allows the user to customize a toolbar so that the toolbar provides convenient access to the commands that are most frequently used by the user. In addition, these programs support multiple toolbars that can be turned on and off, thereby providing the user with the option of viewing two or more toolbars simultaneously. In some prior art systems, the process of customizing or manipulating toolbars requires use of a dialog box that displays a list

of commands available for the toolbar. The dialog box also can display a list of available toolbars that can be displayed for the application. The user can then customize the toolbar by selecting which controls the user wants displayed.

Now, a user typically interacts with a GUI by using a pointing device (e.g., a mouse) to position a pointer or cursor over an object and "clicking" on the object. For example, a drag and drop ("DAD") operation may be initiated by selection from a toolbar or by selecting an object within a digital image. In a typical DAD operation, a pointing device is used to select an object (e.g. text, icons, graphical objects, etc.) under a cursor and then "drag" the selected object to a different location or orientation on a display screen. The user may then "drop" or release the object at a desired new location or orientation indicated by the position of the cursor. Selecting is usually a first step, generally initiated by holding down a button associated with the pointing device (e.g., a mouse button) and gesturing with the pointing device to indicate the bounds of the object to be selected (as in text selection), or simply by "clicking" on the object under the cursor (as in graphical image or icon selection). Selection is typically indicated by a change in the visual display of the selected object (e.g., by using reverse video, displaying a frame around the object, displaying selection handles around the object, etc.). Dragging is usually a separate step distinct from selection, and is usually initiated by clicking on a selected object and holding a control button down (e.g., holding a mouse button in a depressed state). The object is then dragged while holding the control button. However, in some applications, initiating dragging also selects the object under the cursor. The operation is completed by dropping the selected object.

For many applications, a drag operation may be used to initiate several possible functions relative to an object. For example, in a text application, a selected section of text may be moved or copied by a drag operation. Normally, if multiple functions are possible, one such function (e.g., moving) is a "default" function of a drag operation, while the other functions must be selected by some combination of modifiers (e.g., pressing keys like "SHIFT", "ALT", and "CTRL") while operating the pointing device. In some applications, after completing the drag operation, a menu automatically pops up to allow a user to choose a specific "drop" function. In other applications, such as that described in U.S. Patent No. 6,246,411 to Strauss, a user may select among multiple functions during a drag operation using a toolbar, thus allowing the user to change a gesture after it has begun.

One problem with present DAD methods such as that described by Strauss is that a user may have difficulty selecting the object to be dragged or the location where that object is to be a dropped. Thus, a user may have to repeat the DAD operation several times in order to achieve the desired result. In other words, while present DAD methods may provide a user with a desired image after several iterations, these methods do not provide for the accurate selection and positioning of the desired object at the outset. Thus, and especially for large image presentations such as digital maps, a user may have to repeat the DAD operation several times in order to accurately select or position the desired object. For example, while a user may use well-known "panning" and "zooming" tools to view a desired object in an original image in order to reposition that object, in doing so, the relative location of the new position for that object may be lost to the user or the user may find it difficult to determine what portion of the original image is being observed. Thus, while the user may have gained a detailed view of a region of the original image that is of interest, the user may lose sight of the context within which that region is positioned. This is an example of what is often referred to as the “screen real estate problem”.

A need therefore exists for an improved method and system for performing accurate drag and drop operations on digital images. Consequently, it is an object of the present invention to obviate or mitigate at least some of the above mentioned disadvantages.

SUMMARY OF THE INVENTION

According to one aspect of the invention, there is provided a method for positioning a selected object in a computer generated original image on a display, comprising the steps of: distorting the original image to produce a distorted region for the object; dragging the object and the distorted region to a desired position; and, dropping the object at the desired position, whereby the object is accurately positioned.

Preferably, the step of distorting further includes the steps of: creating a lens surface for the distorted region; and, transforming the original image by applying a distortion function defining the lens surface to the original image.

Preferably, the step of creating further includes the step of displaying a GUI over the distorted region for adjusting the lens surface.

Preferably, the lens surface includes a focal region and a base region and the GUI includes: a slide bar icon for adjusting a magnification for the lens surface; a slide bar icon for adjusting a degree of scooping for the lens surface; a bounding rectangle icon with at least one handle icon for adjusting a size and a shape for the focal region; a bounding rectangle icon with at least one handle icon for adjusting a size and a shape for the base region; a move icon for adjusting a location for the lens surface within the original image; a pickup icon for adjusting a location for the base region within the original image; and, a fold icon for adjusting a location for the focal region relative to the base region.

Preferably, the GUI further includes an attached toolbar and the toolbar includes function selection icons and/or function status icons.

Preferably, the steps of dragging, dropping, and adjusting are performed by moving a cursor on the display with a pointing device; the cursor is an icon; and, the pointing device is a mouse.

Preferably, the distorted region is on or overlaps the object.

Preferably, the object is a selection from the original image, an icon, a text selection, or a selection from an external source. Preferably, the external source is an image other than the original image.

Preferably, the step of dragging further includes the step of cutting the object from the original image.

Preferably, the step of dropping further includes the step of pasting the object into the original image.

Preferably, the display is a touchscreen display of a photograph processing workstation or kiosk.

Preferably, the toolbar includes an icon representing the object.

Preferably, the toolbar is transparent, thereby allowing observation of the original image through the toolbar. The toolbar may also be translucent.

According to another aspect of the invention, there is provided a method for manipulating a presentation of a region-of-interest within visual information on a display screen of a computer, the region-of-interest including a focal region and a base region, the method comprising the steps of: displaying a toolbar over the region-of-interest for selecting at least one parameter for transforming at least one of the region-of-interest, the focal region, and the base region; selecting the at least one parameter from the toolbar with a pointing device; transforming the visual information in accordance with a predetermined distortion function and the at least one parameter to produce transformed visual information; and, displaying the transformed visual information on the display screen.

Preferably, the at least one parameter includes: a magnification for the region-of-interest; a size for the focal region; a size for the base region; a shape for the focal region; a shape for the base region; a location for the region-of-interest within the visual information; a location for the base region within the visual information; a location for the focal region relative to the base region; and, a degree of scooping between the focal and base regions.

Preferably, the toolbar includes at least one lens icon for selecting the at least one parameter. Preferably, the at least one lens icon represents the transformed visual information. Preferably, the at least one lens icon includes a pyramidal lens icon, a circular based lens icon, and a circular focused lens icon.

Preferably, the toolbar includes: a pointer icon for selecting points in the visual information; a hand tool icon for selecting a view area in the visual information; a zoom tool icon for zooming into or away from the region-of-interest; a measuring tool icon for initiating a measurement function; a help tool icon for initiating a user help function; a continuation arrow icon for indicating and scrolling additional toolbar icons into view; a delete icon for deleting the presentation from the transformed visual information; a printer icon for selecting and indicating a status of a print function; a floppy disk icon for selecting and indicating a status of a save function; a redo icon for selecting a redo function; an undo icon for selecting an undo function; a resize base icon for selecting a predefined base region resizing function; and, a resize focus icon for selecting a predefined focal region resizing function.

Preferably, the toolbar is a horizontal or vertical toolbar. Preferably, the toolbar is distributed over boundaries of the base and focal regions.

Advantageously, since the magnification at the focus of the distorted region or lens is greater than that at the base of the lens, dragging the lens with the selected object makes it easier for a user to align a point on the selected object with a point at the new location for the object in the digital image or presentation. As the lens with its selected object moves to the new location, the region around the new location is magnified to the same high resolution. As a result, the accuracy of aligning the point on the selected object to a desired point at the new location is improved. In addition, the user is assisted throughout this drag and drop operation by being able to observe the detail in the lens focus in the context of the surrounding presentation.

In addition, by attaching the toolbar to the lens GUI, the controls available through the toolbar are made more easily accessible to a user. As the lens and GUI move, the toolbar moves with them allowing the user easy access to alternate lens applications as the user homes-in on an object or region-of-interest in a presentation. In this way, if the user decides to change lens applications, the toolbar is exactly where the user requires it, that is, near the lens.

BRIEF DESCRIPTION OF THE DRAWINGS

Embodiments of the invention may best be understood by referring to the following description and accompanying drawings. In the description and drawings, like numerals refer to like structures or processes. In the drawings:

FIG. 1 is a graphical representation of the geometry for constructing a three-dimensional (3D) perspective viewing frustum, relative to an x, y, z coordinate system, in accordance with known elastic presentation space graphics technology;

FIG. 2 is a graphical representation of the geometry of a presentation in accordance with known elastic presentation space graphics technology;

FIG. 3 is a block diagram illustrating a data processing system adapted for implementing an embodiment of the invention;

FIG. 4 a partial screen capture illustrating a GUI having lens control elements for user interaction with detail-in-context data presentations in accordance with an embodiment of the invention;

FIG. 5 is a screen capture illustrating a GUI having lens control elements and an attached horizontal toolbar for user interaction with a detail-in-context data presentation in accordance with an embodiment of the invention;

FIG. 6 is a screen capture illustrating a GUI having lens control elements and an attached vertical toolbar for user interaction with a detail-in-context data presentation in accordance with an embodiment of the invention;

FIG. 7 is a screen capture illustrating a GUI having lens control elements and an attached corner toolbar for user interaction with a detail-in-context data presentation in accordance with an embodiment of the invention;

FIG. 8 is a screen capture illustrating a GUI having toolbar icons placed over base and focus resize handle icons for user interaction with a detail-in-context data presentation in accordance with an embodiment of the invention;

FIG. 9 is a screen capture illustrating a selected object in an original image in accordance with an embodiment of the invention;

FIG. 10 is a screen capture illustrating the attachment of a lens to a selected object to produce a detail-in-context presentation in accordance with an embodiment of the invention;

FIG. 11 is a screen capture illustrating a drop and drag operation for a detail-in-context presentation in accordance with an embodiment of the invention; and,

FIG. 12 is a flow chart illustrating a method for positioning a selected object in a computer generated original image on a display in accordance with an embodiment of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

In the following description, numerous specific details are set forth to provide a thorough understanding of the invention. However, it is understood that the invention may be practiced

without these specific details. In other instances, well-known software, circuits, structures and techniques have not been described or shown in detail in order not to obscure the invention. The term “data processing system” is used herein to refer to any machine for processing data, including the computer systems and network arrangements described herein.

The “screen real estate problem” mentioned above generally arises whenever large amounts of information are to be displayed on a display screen of limited size. As discussed, well-known tools to address this problem include panning and zooming. While these tools are suitable for a large number of visual display applications, they become less effective where sections of the visual information are spatially related, such as in maps, three-dimensional representations, and newspapers, for example. In this type of information display, panning and zooming are not as effective as much of the context of the panned or zoomed display may be hidden.

A recent solution to this problem is the application of “detail-in-context” presentation techniques. Detail-in-context is the magnification of a particular region-of-interest (the “focal region” or "detail") in a data presentation while preserving visibility of the surrounding information (the "context"). This technique has applicability to the display of large surface area media (e.g. digital maps) on computer screens of variable size including graphics workstations, laptop computers, personal digital assistants ("PDAs"), and cell phones.

In the detail-in-context discourse, differentiation is often made between the terms “representation” and “presentation”. A representation is a formal system, or mapping, for specifying raw information or data that is stored in a computer or data processing system. For example, a digital map of a city is a representation of raw data including street names and the relative geographic location of streets and utilities. Such a representation may be displayed visually on a computer screen or printed on paper. On the other hand, a presentation is a spatial organization of a given representation that is appropriate for the task at hand. Thus, a presentation of a representation organizes such things as the point of view and the relative emphasis of different parts or regions of the representation. For example, a digital map of a city may be presented with a region magnified to reveal street names.

In general, a detail-in-context presentation may be considered as a distorted view (or distortion) of a portion of the original representation where the distortion is the result of the application of a

“lens” like distortion function to the original representation. A detailed review of various detail-in-context presentation techniques such as “Elastic Presentation Space” (“EPS”) (or “Pliable Display Technology” (“PDT”)) may be found in a publication by Marianne S. T. Carpendale, entitled “A Framework for Elastic Presentation Space” (Carpendale, Marianne S. T., *A Framework for Elastic Presentation Space* (Burnaby, British Columbia: Simon Fraser University, 1999)), and incorporated herein by reference.

In general, detail-in-context data presentations are characterized by magnification of areas of an image where detail is desired, in combination with compression of a restricted range of areas of the remaining information (i.e. the context), the result typically giving the appearance of a lens having been applied to the display surface. Using the techniques described by Carpendale, points in a representation are displaced in three dimensions and a perspective projection is used to display the points on a two-dimensional presentation display. Thus, when a lens is applied to a two-dimensional continuous surface representation, for example, the resulting presentation appears to be three-dimensional. In other words, the lens transformation appears to have stretched the continuous surface in a third dimension. In EPS graphics technology, a two-dimensional visual representation is placed onto a surface; this surface is placed in three-dimensional space; the surface, containing the representation, is viewed through perspective projection; and the surface is manipulated to effect the reorganization of image details. The presentation transformation is separated into two steps: surface manipulation or distortion and perspective projection.

FIG. 1 is a graphical representation **100** of the geometry for constructing a three-dimensional ("3D") perspective viewing frustum **220**, relative to an x, y, z coordinate system, in accordance with known elastic presentation space (EPS) graphics technology. In EPS technology, detail-in-context views of two-dimensional ("2D") visual representations are created with sight-line aligned distortions of a 2D information presentation surface within a 3D perspective viewing frustum **220**. In EPS, magnification of regions of interest and the accompanying compression of the contextual region to accommodate this change in scale are produced by the movement of regions of the surface towards the viewpoint ("VP") **240** located at the apex of the pyramidal shape **220** containing the frustum. The process of projecting these transformed layouts via a perspective projection results in a new 2D layout which includes the zoomed and compressed

regions. The use of the third dimension and perspective distortion to provide magnification in EPS provides a meaningful metaphor for the process of distorting the information presentation surface. The 3D manipulation of the information presentation surface in such a system is an intermediate step in the process of creating a new 2D layout of the information.

FIG. 2 is a graphical representation **200** of the geometry of a presentation in accordance with known EPS graphics technology. EPS graphics technology employs viewer-aligned perspective projections to produce detail-in-context presentations in a reference view plane **201** which may be viewed on a display. Undistorted 2D data points are located in a basal plane **210** of a 3D perspective viewing volume or frustum **220** which is defined by extreme rays **221** and **222** and the basal plane **210**. The VP **240** is generally located above the centre point of the basal plane **210** and reference view plane ("RVP") **201**. Points in the basal plane **210** are displaced upward onto a distorted surface **230** which is defined by a general 3D distortion function (i.e. a detail-in-context distortion basis function). The direction of the viewer-aligned perspective projection corresponding to the distorted surface **230** is indicated by the line FPo - FP **231** drawn from a point FPo **232** in the basal plane **210** through the point FP **233** which corresponds to the focus or focal region or focal point of the distorted surface **230**.

EPS is applicable to multidimensional data and is well suited to implementation on a computer for dynamic detail-in-context display on an electronic display surface such as a monitor. In the case of two dimensional data, EPS is typically characterized by magnification of areas of an image where detail is desired **233**, in combination with compression of a restricted range of areas of the remaining information (i.e. the context) **234**, the end result typically giving the appearance of a lens **230** having been applied to the display surface. The areas of the lens **230** where compression occurs may be referred to as the "shoulder" **234** of the lens **230**. The area of the representation transformed by the lens may be referred to as the "lensed area". The lensed area thus includes the focal region and the shoulder. To reiterate, the source image or representation to be viewed is located in the basal plane **210**. Magnification **233** and compression **234** are achieved through elevating elements of the source image relative to the basal plane **210**, and then projecting the resultant distorted surface onto the reference view plane **201**. EPS performs detail-in-context presentation of n-dimensional data through the use of a procedure wherein the data is mapped into a region in an (n+1) dimensional space, manipulated through perspective

projections in the (n+1) dimensional space, and then finally transformed back into n-dimensional space for presentation. EPS has numerous advantages over conventional zoom, pan, and scroll technologies, including the capability of preserving the visibility of information outside **234** the local region of interest **233**.

For example, and referring to FIGS. 1 and 2, in two dimensions, EPS can be implemented through the projection of an image onto a reference plane **201** in the following manner. The source image or representation is located on a basal plane **210**, and those regions of interest **233** of the image for which magnification is desired are elevated so as to move them closer to a reference plane situated between the reference viewpoint **240** and the reference view plane **201**. Magnification of the focal region **233** closest to the RVP **201** varies inversely with distance from the RVP **201**. As shown in FIGS. 1 and 2, compression of regions **234** outside the focal region **233** is a function of both distance from the RVP **201**, and the gradient of the function describing the vertical distance from the RVP **201** with respect to horizontal distance from the focal region **233**. The resultant combination of magnification **233** and compression **234** of the image as seen from the reference viewpoint **240** results in a lens-like effect similar to that of a magnifying glass applied to the image. Hence, the various functions used to vary the magnification and compression of the source image via vertical displacement from the basal plane **210** are described as lenses, lens types, or lens functions. Lens functions that describe basic lens types with point and circular focal regions, as well as certain more complex lenses and advanced capabilities such as folding, have previously been described by Carpendale.

System. FIG. 3 is a block diagram of a data processing system **300** adapted to implement an embodiment of the invention. The data processing system is suitable for implementing EPS technology, for displaying detail-in-context presentations of representations, and for performing drag and drop ("DAD") operations in conjunction with a detail-in-context graphical user interface ("GUI") **400**, as described below. The data processing system **300** includes an input device **310**, a central processing unit or CPU **320**, memory **330**, and a display **340**. The input device **310** may include a keyboard, mouse, trackball, or similar device. The CPU **320** may include dedicated coprocessors and memory devices. The memory **330** may include RAM, ROM, databases, or disk devices. And, the display **340** may include a computer screen, terminal device, or a hardcopy producing output device such as a printer or plotter. The data processing

system **300** has stored therein data representing sequences of instructions which when executed cause the method described herein to be performed. Of course, the data processing system **300** may contain additional software and hardware a description of which is not necessary for understanding the invention.

GUI with Lens Control Elements. As mentioned, detail-in-context presentations of data using techniques such as pliable surfaces, as described by Carpendale, are useful in presenting large amounts of information on limited-size display surfaces. Detail-in-context views allow magnification of a particular region-of-interest (the "focal region") **233** in a data presentation while preserving visibility of the surrounding information **210**. In the following, a GUI **400** is described having lens control elements that can be implemented in software and applied to DAD operations and to the control of detail-in-context data presentations. The software can be loaded into and run by the data processing system **300** of FIG. 3.

FIG. 4 is a partial screen capture illustrating a GUI **400** having lens control elements for user interaction with detail-in-context data presentations in accordance with an embodiment of the invention. Detail-in-context data presentations are characterized by magnification of areas of an image where detail is desired, in combination with compression of a restricted range of areas of the remaining information (i.e. the context), the end result typically giving the appearance of a lens having been applied to the display screen surface. This lens **410** includes a "focal region" **420** having high magnification, a surrounding "shoulder region" **430** where information is typically visibly compressed, and a "base" **412** surrounding the shoulder region **430** and defining the extent of the lens **410**. In FIG. 4, the lens **410** is shown with a circular shaped base **412** (or outline) and with a focal region **420** lying near the center of the lens **410**. However, the lens **410** and focal region **420** may have any desired shape. For example, in FIG. 5, the lens **410** has a pyramid shape with a flat top **420** and trapezoidal shoulders **430**. As mentioned above, the base of the lens **412** may be coextensive with the focal region **420**.

In general, the GUI **400** has lens control elements that, in combination, provide for the interactive control of the lens **410**. The effective control of the characteristics of the lens **410** by a user (i.e. dynamic interaction with a detail-in-context lens) is advantageous. At any given time, one or more of these lens control elements may be made visible to the user on the display surface

340 by appearing as overlay icons on the lens **410**. Interaction with each element is performed via the motion of an input or pointing device **310** (e.g. mouse), with the motion resulting in an appropriate change in the corresponding lens characteristic. As will be described, selection of which lens control element is actively controlled by the motion of the pointing device **310** at any given time is determined by the proximity of the icon representing the pointing device **310** (e.g. cursor) on the display surface **340** to the appropriate component of the lens **410**. For example, "dragging" of the pointing device at the periphery of the bounding rectangle of the lens base **412** causes a corresponding change in the size of the lens **410** (i.e. "resizing"). Thus, the GUI **400** provides the user with a visual representation of which lens control element is being adjusted through the display of one or more corresponding icons.

For ease of understanding, the following discussion will be in the context of using a two-dimensional pointing device **310** that is a mouse, but it will be understood that the invention may be practiced with other 2-D or 3-D (or even greater numbers of dimensions) pointing devices including a trackball and keyboard.

A mouse **310** controls the position of a cursor icon **401** that is displayed on the display screen **340**. The cursor **401** is moved by moving the mouse **310** over a flat surface, such as the top of a desk, in the desired direction of movement of the cursor **401**. Thus, the two-dimensional movement of the mouse **310** on the flat surface translates into a corresponding two-dimensional movement of the cursor **401** on the display screen **340**.

A mouse **310** typically has one or more finger actuated control buttons (i.e. mouse buttons). While the mouse buttons can be used for different functions such as selecting a menu option pointed at by the cursor **401**, the disclosed invention may use a single mouse button to "select" a lens **410** and to trace the movement of the cursor **401** along a desired path. Specifically, to select a lens **410**, the cursor **401** is first located within the extent of the lens **410**. In other words, the cursor **401** is "pointed" at the lens **410**. Next, the mouse button is depressed and released. That is, the mouse button is "clicked". Selection is thus a point and click operation. To trace the movement of the cursor **401**, the cursor **401** is located at the desired starting location, the mouse button is depressed to signal the computer **320** to activate a lens control element, and the mouse **310** is moved while maintaining the button depressed. After the desired path has been traced, the

mouse button is released. This procedure is often referred to as "clicking" and "dragging" (i.e. a click and drag operation). It will be understood that a predetermined key on a keyboard **310** could also be used to activate a mouse click or drag. In the following, the term "clicking" will refer to the depression of a mouse button indicating a selection by the user and the term "dragging" will refer to the subsequent motion of the mouse **310** and cursor **401** without the release of the mouse button.

The GUI **400** may include the following lens control elements: move, pickup, resize base, resize focus, fold, magnify, and scoop. Each of these lens control elements has at least one lens control icon or alternate cursor icon associated with it. In general, when a lens **410** is selected by a user through a point and click operation, the following lens control icons may be displayed over the lens **410**: pickup icon **450**, base outline icon **412**, base bounding rectangle icon **411**, focal region bounding rectangle icon **421**, handle icons **481**, **482**, **491**, **492** (see FIG. 5), magnify slide bar icon **440**, and scoop slide bar icon **540** (see FIG. 5). Typically, these icons are displayed simultaneously after selection of the lens **410**. In addition, when the cursor **401** is located within the extent of a selected lens **410**, an alternate cursor icon **460**, **470**, **480**, **490** may be displayed over the lens **410** to replace the cursor **401** or may be displayed in combination with the cursor **401**. These lens control elements, corresponding icons, and their effects on the characteristics of a lens **410** are described below with reference to FIG. 4.

In general, when a lens **410** is selected by a point and click operation, bounding rectangle icons **411**, **421** are displayed surrounding the base **412** and focal region **420** of the selected lens **410** to indicate that the lens **410** has been selected. With respect to the bounding rectangles **411**, **421** one might view them as glass windows enclosing the lens base **412** and focal region **420**, respectively. The bounding rectangles **411**, **421** include handle icons **481**, **482**, **491**, **492** allowing for direct manipulation of the enclosed base **412** and focal region **420** as will be explained below. Thus, the bounding rectangles **411**, **421** not only inform the user that the lens **410** has been selected, but also provide the user with indications as to what manipulation operations might be possible for the selected lens **410** though use of the displayed handles **481**, **482**, **491**, **492**. Note that it is well within the scope of the present invention to provide a bounding region having a shape other than generally rectangular. Such a bounding region could be of any of a great

number of shapes including oblong, oval, ovoid, conical, cubic, cylindrical, polyhedral, spherical, etc.

Moreover, the cursor **401** provides a visual cue indicating the nature of an available lens control element. As such, the cursor **401** will generally change in form by simply pointing to a different lens control icon **450, 412, 411, 421, 481, 482, 491, 492, 440, 540**. For example, when resizing the base **412** of a lens **410** using a corner handle **491**, the cursor **401** will change form to a resize icon **490** once it is pointed at (i.e. positioned over) the corner handle **491**. The cursor **401** will remain in the form of the resize icon **490** until the cursor **401** has been moved away from the corner handle **491**.

Move. Lateral movement of a lens **410** is provided by the move lens control element of the GUI **400**. This functionality is accomplished by the user first selecting the lens **410** through a point and click operation. Then, the user points to a point within the lens **410** that is other than a point lying on a lens control icon **450, 412, 411, 421, 481, 482, 491, 492, 440, 540**. When the cursor **401** is so located, a move icon **460** is displayed over the lens **410** to replace the cursor **401** or may be displayed in combination with the cursor **401**. The move icon **460** not only informs the user that the lens **410** may be moved, but also provides the user with indications as to what movement operations are possible for the selected lens **410**. For example, the move icon **460** may include arrowheads indicating up, down, left, and right motion. Next, the lens **410** is moved by a click and drag operation in which the user clicks and drags the lens **410** to the desired position on the screen **340** and then releases the mouse button **310**. The lens **410** is locked in its new position until a further pickup and move operation is performed.

Pickup. Lateral movement of a lens **410** is also provided by the pickup lens control element of the GUI. This functionality is accomplished by the user first selecting the lens **410** through a point and click operation. As mentioned above, when the lens **410** is selected a pickup icon **450** is displayed over the lens **410** at the cursor location (e.g. near the centre of the lens **410**). Typically, the pickup icon **450** will be a crosshairs. In addition, a base outline **412** is displayed over the lens **410** representing the base **412** of the lens **410**. The crosshairs **450** and lens outline **412** not only inform the user that the lens has been selected, but also provides the user with an indication as to the pickup operation that is possible for the selected lens **410**. Next, the user

points at the crosshairs **450** with the cursor **401**. Then, the lens outline **412** is moved by a click and drag operation in which the user clicks and drags the crosshairs **450** to the desired position on the screen **340** and then releases the mouse button **310**. The full lens **410** is then moved to the new position and is locked there until a further pickup operation is performed. In contrast to the move operation described above, with the pickup operation, it is the outline **412** of the lens **410** that the user repositions rather than the full lens **410**.

Resize Base. Resizing of the base **412** (or outline) of a lens **410** is provided by the resize base lens control element of the GUI. After the lens **410** is selected, a bounding rectangle icon **411** is displayed surrounding the base **412**. The bounding rectangle **411** includes handles **491**. These handles **491** can be used to stretch the base **412** taller or shorter, wider or narrower, or proportionally larger or smaller. The corner handles **491** will keep the proportions the same while changing the size. The middle handles **492** (see FIG. 5) will make the base **412** taller or shorter, wider or narrower. Resizing the base **412** by the corner handles **491** will keep the base **412** in proportion. Resizing the base **412** by the middle handles **492** will change the proportions of the base **412**. That is, the middle handles **492** change the aspect ratio of the base **412** (i.e. the ratio between the height and the width of the bounding rectangle **411** of the base **412**). When a user points at a handle **491** with the cursor **401** a resize icon **490** may be displayed over the handle **491** to replace the cursor **401** or may be displayed in combination with the cursor **401**. The resize icon **490** not only informs the user that the handle **491** may be selected, but also provides the user with indications as to the resizing operations that are possible with the selected handle. For example, the resize icon **490** for a corner handle **491** may include arrows indicating proportional resizing. The resize icon (not shown) for a middle handle **492** may include arrows indicating width resizing or height resizing. After pointing at the desired handle **491**, **492**, the user would click and drag the handle **491**, **492** until the desired shape and size for the base **412** is reached. Once the desired shape and size are reached, the user would release the mouse button **310**. The base **412** of the lens **410** is then locked in its new size and shape until a further base resize operation is performed.

Resize Focus. Resizing of the focal region **420** of a lens **410** is provided by the resize focus lens control element of the GUI. After the lens **410** is selected, a bounding rectangle icon **421** is displayed surrounding the focal region **420**. The bounding rectangle **421** includes handles 481,

482. These handles **481, 482** can be used to stretch the focal region **420** taller or shorter, wider or narrower, or proportionally larger or smaller. The corner handles **481** will keep the proportions the same while changing the size. The middle handles **482** will make the focal region **420** taller or shorter, wider or narrower. Resizing the focal region **420** by the corner handles **481** will keep the focal region **420** in proportion. Resizing the focal region **420** by the middle handles **482** will change the proportions of the focal region **420**. That is, the middle handles **482** change the aspect ratio of the focal region **420** (i.e. the ratio between the height and the width of the bounding rectangle **421** of the focal region **420**). When a user points at a handle **481, 482** with the cursor **401** a resize icon **480** may be displayed over the handle **481, 482** to replace the cursor **401** or may be displayed in combination with the cursor **401**. The resize icon **480** not only informs the user that a handle **481, 482** may be selected, but also provides the user with indications as to the resizing operations that are possible with the selected handle. For example, the resize icon **480** for a corner handle **481** may include arrows indicating proportional resizing. The resize icon **480** for a middle handle **482** may include arrows indicating width resizing or height resizing. After pointing at the desired handle **481, 482**, the user would click and drag the handle **481, 482** until the desired shape and size for the focal region **420** is reached. Once the desired shape and size are reached, the user would release the mouse button **310**. The focal region **420** is then locked in its new size and shape until a further focus resize operation is performed.

Fold. Folding of the focal region **420** of a lens **410** is provided by the fold control element of the GUI. In general, control of the degree and direction of folding (i.e. skewing of the viewer aligned vector **231** as described by Carpendale) is accomplished by a click and drag operation on a point **471**, other than a handle **481, 482,** on the bounding rectangle **421** surrounding the focal region **420**. The direction of folding is determined by the direction in which the point **471** is dragged. The degree of folding is determined by the magnitude of the translation of the cursor **401** during the drag. In general, the direction and degree of folding corresponds to the relative displacement of the focus **420** with respect to the lens base **410**. In other words, and referring to FIG. 2, the direction and degree of folding corresponds to the displacement of the point FP **233** relative to the point FPo **232**, where the vector joining the points FPo **232** and FP **233** defines the viewer aligned vector **231**. In particular, after the lens **410** is selected, a bounding rectangle icon **421** is displayed surrounding the focal region **420**. The bounding rectangle **421** includes handles **481,**

482. When a user points at a point **471**, other than a handle **481**, **482**, on the bounding rectangle **421** surrounding the focal region **420** with the cursor **401**, a fold icon **470** may be displayed over the point **471** to replace the cursor **401** or may be displayed in combination with the cursor **401**. The fold icon **470** not only informs the user that a point **471** on the bounding rectangle **421** may be selected, but also provides the user with indications as to what fold operations are possible. For example, the fold icon **470** may include arrowheads indicating up, down, left, and right motion. By choosing a point **471**, other than a handle **481**, **482**, on the bounding rectangle **421** a user may control the degree and direction of folding. To control the direction of folding, the user would click on the point **471** and drag in the desired direction of folding. To control the degree of folding, the user would drag to a greater or lesser degree in the desired direction of folding. Once the desired direction and degree of folding is reached, the user would release the mouse button **310**. The lens **410** is then locked with the selected fold until a further fold operation is performed.

Magnify. Magnification of the lens **410** is provided by the magnify lens control element of the GUI. After the lens **410** is selected, the magnify control is presented to the user as a slide bar icon **440** near or adjacent to the lens **410** and typically to one side of the lens **410**. Sliding the bar **441** of the slide bar **440** results in a proportional change in the magnification of the lens **410**. The slide bar **440** not only informs the user that magnification of the lens **410** may be selected, but also provides the user with an indication as to what level of magnification is possible. The slide bar **440** includes a bar **441** that may be slid up and down, or left and right, to adjust and indicate the level of magnification. To control the level of magnification, the user would click on the bar **441** of the slide bar **440** and drag in the direction of desired magnification level. Once the desired level of magnification is reached, the user would release the mouse button **310**. The lens **410** is then locked with the selected magnification until a further magnification operation is performed. In general, the focal region **420** is an area of the lens **410** having constant magnification (i.e. if the focal region is a plane). Again referring to FIGS. 1 and 2, magnification of the focal region **420**, **233** varies inversely with the distance from the focal region **420**, **233** to the reference view plane (RVP) **201**. Magnification of areas lying in the shoulder region **430** of the lens **410** also varies inversely with their distance from the RVP **201**. Thus, magnification of areas lying in the shoulder region **430** will range from unity at the base **412** to the level of magnification of the focal region **420**.

Scoop. The concavity or "scoop" of the shoulder region **430** of the lens **410** is provided by the scoop lens control element of the GUI. After the lens **410** is selected, the scoop control is presented to the user as a slide bar icon **540** (see FIG. 5) near or adjacent to the lens **410** and typically below the lens **410**. Sliding the bar **541** of the slide bar **540** results in a proportional change in the concavity or scoop of the shoulder region **430** of the lens **410**. The slide bar **540** not only informs the user that the shape of the shoulder region **430** of the lens **410** may be selected, but also provides the user with an indication as to what degree of shaping is possible. The slide bar **540** includes a bar **541** that may be slid left and right, or up and down, to adjust and indicate the degree of scooping. To control the degree of scooping, the user would click on the bar **541** of the slide bar **540** and drag in the direction of desired scooping degree. Once the desired degree of scooping is reached, the user would release the mouse button **310**. The lens **410** is then locked with the selected scoop until a further scooping operation is performed.

Icon Hiding. Advantageously, a user may choose to hide one or more lens control icons **450**, **412**, **411**, **421**, **481**, **482**, **491**, **492**, **440**, **540** shown in FIGS. 4 and 5 from view so as not to impede the user's view of the image within the lens **410**. This may be helpful, for example, during a move or drag and drop operation. A user may select this option through means such as a menu or lens property dialog box.

GUI with Lens Control Elements and Attached Toolbar. Detail-in-context lenses may be used in a variety of applications. Using the GUI **400** described above, a user may change the manner in which information is displayed or presented. For example, the user may change the shape of the lens **410** (e.g. from pyramid shaped to cone shaped). Detail-in-context lens may also be used for editing applications. For example, a user may change the colour of a pixel, or add a label to source data, using a detail-in-context lens as a selection device. Detail-in-context lens may be used for more complex editing applications such as DAD operations as will be described in more detail below. Such detail-in-context lens applications or modes may be chosen by a user via keyboard commands, pull-down menu, or toolbar.

FIG. 5 is a screen capture illustrating a GUI **500** having lens control elements and an attached horizontal toolbar **510** for user interaction with a detail-in-context data presentation **550** in accordance with an embodiment of the invention. In FIG. 5, the toolbar **510** is located above the

lens **410**. The toolbar **510** includes a number of icons **561, 562, 563, 564, 565, 566** for selecting an application for the lens **410** and/or for providing related functions. In FIG. 5, the toolbar **510** includes a pointer icon **561** for selecting points in the presentation **550** using the lens **410**, a hand tool icon **562** for selecting a view area for the presentation **550**, a zoom tool icon **563** for zooming into or away from the region-of-interest **420** or presentation **550** (see the Applicant's co-pending Canadian Patent Application No. 2,350,342, which is incorporated herein by reference), a measuring tool icon **564** for initiating a measurement operation (see the Applicant's co-pending Canadian Patent Application Nos. 2,393,708 and 2,394,119, which are incorporated herein by reference), a help tool icon **565** for initiating a user help function as is known in the art, and a continuation arrow icon **566** for indicating the existence of and scrolling additional toolbar icons into view.

FIG. 6 is a screen capture illustrating a GUI **600** having lens control elements and an attached vertical toolbar **610** for user interaction with a detail-in-context data presentation **550** in accordance with an embodiment of the invention. In FIG. 6, the toolbar **610** is located at the side of the lens **410**. Again, the toolbar **610** includes a number of icons **661, 662, 663, 664, 665, 666, 566** for selecting an application for the lens **410** and/or for providing related functions. The toolbar **610** includes a pyramid lens icon **661** for choosing a pyramid shaped lens **410**, a circular based lens icon **662** for choosing a lens **410** with a circular base **412**, a circular focus icon **663** for choosing a lens **410** with a circular shaped focus **420**, a delete icon **664** for deleting a lens from the presentation **550**, and the continuation arrow icon **566** for indicating the existence of and scrolling additional toolbar icons into view. Additional icons **665, 666** may be included for additional functions, as needed, or may be included as reduced-sized representations of the data or objects to be copied, cut, or placed.

FIG. 7 is a screen capture illustrating a GUI **700** having lens control elements and an attached corner toolbar **710** for user interaction with a detail-in-context data presentation **550** in accordance with an embodiment of the invention. In FIG. 7, the toolbar **710** is located at a corner of the lens **410**. Again, the toolbar **710** includes a number of icons **661, 662, 663, 664, 566** for selecting an application for the lens **410** and/or for providing related functions. The toolbar **710** includes a pyramid lens icon **661** for choosing a pyramid shaped lens **410**, a circular based lens icon **662** for choosing a lens **410** with a circular base **412**, a circular focus icon **663** for choosing

a lens **410** with a circular shaped focus **420**, a delete icon **664** for deleting a lens from the presentation **550**, and the continuation arrow icon **566** for indicating the existence of and scrolling additional toolbar icons into view.

Advantageously, by attaching the toolbar **510, 610, 710** to the lens GUI **400**, the controls available through the toolbar **510, 610, 710** are made more easily accessible to a user. As the lens **410** and GUI **500, 600, 700** move, the toolbar **510 610, 710** moves with them allowing the user easy access to alternate lens applications as the user homes-in on a region-of-interest **420** in a presentation **550**. In this way, if the user decides to change lens applications, the toolbar **510, 610, 710** is exactly where the user requires it, that is, near the lens **410**.

According to one embodiment, the toolbar **510, 610, 710** need not be visible at all times. An icon, for example the continuation arrow icon **566**, may be used to toggle the visibility of the toolbar **510, 610, 710** on and off. Alternatively, the toolbar **510, 610, 710** may be transparent along with the rest of the GUI **500, 600, 700** as illustrated in FIGS. 5, 6, and 7. According to another embodiment, the toolbar **510, 610, 710** need not be docked or attached to the lens **410** at all times. Rather, an icon (not shown) on the toolbar **510, 610, 710** may be clicked to toggle the toolbar from a docked to a floating state. According to another embodiment, the toolbar **510, 610, 710** or its icons may be resized. For example, the toolbar icons **561, 562, 563, 564, 565, 566** may be presented at a larger size as the user directs the cursor **401** toward them. According to another embodiment, the position of the toolbar **510, 610, 710** may vary with the position of the lens **410** on the display screen **340**. For example, if a presentation **550** has a lens **410** positioned at the top-left corner of the screen **340**, then the toolbar **710** may be automatically presented at the bottom-right corner of the lens **410** as shown in FIG. 7. According to another embodiment, an icon (not shown) on the toolbar **510, 610, 710** may be clicked to toggle the location of the toolbar about the lens **410**. According to another embodiment, the toolbar **510, 610, 710** may be manually or automatically resized. According to another embodiment, the toolbar icons may represent applications that are currently running. By clicking on a toolbar icon, the user is able to switch from a first application to another from within the first application. According to another embodiment, the toolbar icons may be used to indicate the status of applications that are currently running. For example, a printer icon **861** (see FIG. 8) may indicate that printing is in

progress. Another icon (not shown) may indicate that retrieving of high resolution data through the lens **410** is in progress.

FIG. 8 is a screen capture illustrating a GUI **800** having toolbar icons **561, 562, 563, 564, 565, 566, 661, 662, 664, 861, 862, 863, 864, 865, 866** placed over base and focus resize handle icons **481, 482, 491, 492** for user interaction with a detail-in-context data presentation **550** in accordance with an embodiment of the invention. The GUI **800** includes a number of toolbar icons for selecting an application for the lens **410** and/or for providing related functions. The toolbar icons include a printer icon **861** for selecting or indicating the status of a print application, a floppy disk icon **863** for selecting or indicating the status of a save application, redo/undo icons **846** for selecting redo and undo applications, a resize base icon **865** for selecting a predefined base resizing application, and a resize focus icon **866** for selecting a predefined focus resizing operation. An additional icon **862** may be included for an additional function, as needed, or may be included as a reduced-sized representation of the data or objects to be copied, cut, or placed. According to one embodiment, the toolbar icons may be distributed along the bounding rectangles **411, 421** of the base **412** and focus **420** of the lens **410** rather than being placed over the base and focus resize handle icons **481, 482, 491, 492**. According to another embodiment, an icon (not shown) may be clicked to toggle the location of the toolbar icons from over the base and focus resize handle icons **481, 482, 491, 492** to a toolbar **510, 610, 710** located adjacent to the lens **410**. Advantageously, by placing toolbar icons **561, 562, 563, 564, 565, 566, 661, 662, 664, 861, 862, 863, 864, 865, 866** over base and focus resize handle icons **481, 482, 491, 492**, the visibility of the presentation **550** to the user may be improved for some applications. Again, other icons may be added to represent other functions or to represent data to be cut, copied, or placed.

Dragging and Dropping with Detail-In-Context Lenses. Now, in accordance with the present invention, detail-in-context data viewing techniques are applied to DAD operations in digital image presentations. Detail-in-context data viewing techniques allow a user to view multiple levels of detail or resolution on one display **340**. The appearance of the data display or presentation is that of one or more virtual lens showing detail **233** within the context of a larger area view **210**. In accordance with the present invention, detail-in-context lenses may be used to perform accurate DAD operations.

As mentioned above, a user typically interacts with a GUI by using a pointing device (e.g., a mouse) **310** to position a pointer or cursor **401** over an object and "clicking" on the object. FIG. 9 is a screen capture illustrating such an object **910** in an original image **900** in accordance with an embodiment of the invention. Thus, a drag and drop ("DAD") operation may be initiated by selection from a toolbar **510, 610, 710, 810** or by selecting an object **910** within an original image **900**. The pointing device (e.g. mouse) **310** is used to select an object **910** (e.g., text, icons, graphical objects, etc.) under a cursor **401** and then "drag" the selected object **910** to a different location or orientation on a display screen **340**. The user may then "drop" or release the object **910** at a desired new location or orientation indicated by the position of the cursor **401**. Selecting may be initiated by holding down a button associated with the pointing device (e.g., a mouse button) **310** and gesturing with the pointing device **310** to indicate the bounds of the object **910** to be selected (as in text selection), or simply by "clicking" on the object **910** under the cursor **401** (as in graphical image or icon selection). Selection may be indicated by a change in the visual display of the selected object **910** (e.g., by using reverse video, displaying a frame around the object, displaying selection handles around the object, etc.). In FIG. 9, the selection of the object **910** is indicated by a dashed line **920** bounding the object **910**.

Once an object **910** is selected, a lens **410** is attached to the object **910**. Any point on the selected object **910** may be chosen to be in the centre of the lens focus **420**. FIG. 10 is a screen capture illustrating the attachment of a lens **410** to a selected object **910** to produce a detail-in-context presentation **905** in accordance with an embodiment of the invention. In FIG. 10, the centre of the focus **420** of the lens **410** is attached at any point **930** (e.g. an end point) of the object **910**. The lens **410** may be configured using its associated GUI **400** in the manner described above. That is, the shape, size, magnification, scoop, and fold for the lens **410** may all be carefully tuned for the selected object **910**. The lens **410** may be configured before attachment to the selected object **910** or after attachment. In addition, the lens **410** may be displayed before the object **910** is selected to aid in the selection of the object **910**.

FIG. 11 is a screen capture illustrating a drop and drag operation for a detail-in-context presentation **905** in accordance with an embodiment of the invention. Having selected and attached a lens **410** to the object **910**, the object **910** may now be dragged to its new location **940**. Dragging may be a separate step distinct from selection and attachment, and may be initiated by

clicking on the selected object **910** and depressing a mouse **310** control button. The object **910** is then dragged from its original position **930** to its new position **940** while holding the control button down. As the object **910** moves, the lens **410** moves with it. Alternatively, the lens **410** may be thought of as carrying the object **910**, that is, the object **910** may be attached to the lens **410** such that as the lens **410** moves, the object **910** moves with it. In an alternative embodiment, initiating dragging also selects the object **910** under the cursor **401** and attaches a lens **410** to it. In another embodiment, selecting the object **910** attaches a lens **410** and initiates dragging. The DAD operation is completed by dropping the selected object **910** at its new location **940**. That is, releasing the mouse button when a selected point on the object **930** is aligned with a desired point **940** at the new location.

Advantageously, since the magnification at the focus **420** of the lens **410** is greater than that at the base of the lens **410**, dragging the lens **410** with the selected object **910** makes it easier for a user to align a point **930** on the selected object **910** with a point **940** at the new location for the object in the presentation **905**. For example, the magnification in the focus **420** may be set to pixel level resolution using the magnification slide bar icon **440**. As the lens **410** with its selected object **910** moves to the new location **940**, the region around the new location **940** is magnified to the same high resolution. As a result, the accuracy of aligning the point **930** on the selected object **910** to a desired point **940** at the new location is improved. In addition, the user is assisted throughout this DAD operation by being able to observe the detail in the lens focus **420** in the context of the surrounding presentation **900**. Finally, once the points **930**, **940** are aligned, the object **910** may be dropped from the lens **410** into its new position.

In operation, the data processing system **300** employs EPS techniques with an input device **310** and GUI **500**, **600**, **700**, **800** for selecting an object **910** and points **930**, **940** to perform a DAD operation for display to a user on a display screen **340**. Data representing an original image **900** or representation is received by the CPU **320** of the data processing system **300**. Using EPS techniques, the CPU **320** processes the data in accordance with instructions received from the user via an input device **310** and GUI **500**, **600**, **700**, **800** to produce a detail-in-context presentation **905**. The presentation **905** is presented to the user on a display screen **340**. It will be understood that the CPU **320** may apply a transformation to the shoulder region **430** surrounding the region-of-interest **420** to affect blending or folding in accordance with EPS technology. For

example, the transformation may map the region-of-interest **420** and/or shoulder region **430** to a predefined lens surface, defined by a transformation or distortion function and having a variety of shapes, using EPS techniques. Or, the lens **410** may be simply coextensive with the region-of-interest **420**. (Blending and folding of lenses in detail-in-context presentations are described in United States Patent Application Publication No. 2002/0044154 which is incorporated herein by reference.)

The lens control elements of the GUI **500, 600, 700, 800** are adjusted by the user via an input device **310** to control the characteristics of the lens **410** in the detail-in-context presentation **905**. Using an input device **310** such as a mouse, a user adjusts parameters of the lens **410** using icons and scroll bars of the GUI **500, 600, 700, 800** that are displayed over the lens on the display screen **340**. The user may also adjust parameters of the image of the full scene **905**. Signals representing input device **310** movements and selections are transmitted to the CPU **320** of the data processing system **300** where they are translated into instructions for lens control.

In FIG. 9, the dashed line **920** indicates the object **910** selected for the DAD operation. In FIG. 10, by moving the lens **410** on the display screen **340** with the lens GUI **500, 600, 700, 800**, the user can locate the focus **420** of the lens **410** over a selected point **930** on the object **910** in the presentation **905**. In FIG. 11, observing the points **930, 940** within the focus **420** of the lens **410** as the user drags the object **910**, the user can decide whether or not the current position of the object **910** is desirable. If the user is satisfied with the current position, the user may drop the object **910**. If the user is dissatisfied with the current position of the object **910**, then the object may be dragged to a new position. Advantageously, by using a detail-in-context lens **410** to select an object **910** or points **930, 940** defining a DAD operation, a user can view a large area **905** (i.e. outside the lens **410**) while focusing in on a smaller area **420** (i.e. inside the focal region **420** of the lens **410**) surrounding the selected object **910** or points **930, 940**. This makes it possible for a user to perform an accurate DAD operation without losing visibility or context of the portion of the original image surrounding the selected object **910**.

Moreover, the lens **410** may be added to the presentation **900** before or after the object **910** is selected. That is, the user may first add a lens **410** to a presentation **900** or the user may move a pre-existing lens into place at, say, a selected point **930** on an object **910**. The lens **410** may be

introduced to the original image **900** to form the presentation **905** through the use of a pull-down menu selection, tool bar icon, etc. The DAD operation may then be activated using a toolbar selection or with a double click on the selected object **910**. Now, as the select object **910** is dragged to its new location or point **940**, the lens **410** is presented over and moves with the selected point **930** on the object **910**. Again, this facilitates the accurate selection of the new position or point **940** defining the DAD operation.

The object **910** may consist of raster-based or vector-based data. In the case of vector-based data, a vector object **910** is attached to the lens **410** and may be aligned with other vector or raster data. The object **910** may also consist of text data which may be attached to the lens **410** and dragged to a new location in the presentation **905**. Other objects **910** such as icons and 3-D objects may also be attached and dragged with the lens **410**. For example, an icon representing a file or an application may be attached to the lens **410** and then dragged to the recycle bin for disposal. Or, a 3-D object such as a chair may be carried by the lens **410** and moved to a new location **940**.

A number of pre-configured lenses **410** with or without attached objects **910** may be saved in the memory **330** of the system **300** for subsequent use. These lenses and objects may be saved in a general toolbar, in a lens toolbar **510**, **610**, **710**, **810**, or as a list of bookmarks in a pull-down menu and may be subsequently recalled and pasted into a presentation **905**. In addition, pre-configured lenses **410** may be assigned names by a user. According to one embodiment, a lens **410** may be saved with more that one object **910** attached to it. In this embodiment, when the lens is moved, all of the attached objects move with it.

As described in the Applicant's co-pending Canadian Patent Application Nos. 2,393,708 and 2,394,119, referred to above, detail-in-context lens may be used for cropping an original image. Similarly, and in accordance with an embodiment of the present invention, the shape of a lens **410** applied to an original image **900** may be used to define a selection or "cut" from the original image. Lenses may be formed as squares, circles, or other shapes and these may be modified on the fly using the GUIs described above to form new shapes. A newly shaped lens may then be used as a "cookie cutter" or copier to cut, move (i.e. drag and drop as described above), and paste objects into a current presentation **905** or a new presentation. After performing a lens shaped cut,

the lens may be extended out beyond the bounds of the cut area to allow improved detail-in-context viewing and DAD operation. Moreover, if the original image includes multiple layers or is three-dimensional, then the cut may also include data from the different layers. That is, the cut may be multi-layer or multi-dimensional. In addition, the lens GUI may include a cut depth control slide bar icon or the like for specifying cut depth.

The present invention my be used in photo kiosk applications. In general, a photo kiosk is specialized workstation connected to high quality printers for processing and printing digital photographic images, typically for a fee paid by the user. Photo kiosks typically have touch-sensitive screens ("touchscreens") for user input. The GUIs described above may be advantageously used in photo kiosks to facilitate user editing of digital images. For example, the lens extent handles **491**, **492** and focus handles **481**, **482** can be used to select the region or regions of a displayed photo which are to be printed or otherwise processed, and then the attached toolbar **510**, **610**, **710**, **810** can be used to select a processing or printing operation to be performed. In one embodiment, the lens extent and focus handles **491**, **491**, **481**, **482** can be overlaid with icons representing the status of a selected operation (e.g. printing, copying, or other processing of the image or of a part of the image). Moreover, the DAD operations described above can be used for adding images (e.g. flowers, hearts, text, background images, "Clipart", etc.) to an original image with enhanced placement accuracy.

Method. FIG. 12 is a flow chart **1200** illustrating a method for positioning a selected object **910** in a computer generated original image **900** on a display **340** in accordance with an embodiment of the invention. At step **1201**, the method starts.

At step **1202**, the original image **900** is distorted to produce a distorted region **410** for the object **910**. This step of distorting may further includes the steps of: creating a lens surface for the distorted region **410**; and, transforming the original image **900** by applying a distortion function defining the lens surface to the original image **900**. The step of creating may further include the step of displaying a GUI **400**, **500**, **600**, **700**, **800** over the distorted region **410** for adjusting the lens surface.

At step **1203**, the object **910** and the distorted region **410** are dragged to a desired position **940**.

At step **1204**, the object **910** is dropped at the desired position **940**. By using the distorted region **410**, the object **910** is accurately positioned.

At step **1205**, the method ends.

Data Carrier Product. The sequences of instructions which when executed cause the method described herein to be performed by the exemplary data processing system of FIG. 3 can be contained in a data carrier product according to one embodiment of the invention. This data carrier product can be loaded into and run by the exemplary data processing system of FIG. 3.

Computer Software Product. The sequences of instructions which when executed cause the method described herein to be performed by the exemplary data processing system of FIG. 3 can be contained in a computer software product according to one embodiment of the invention. This computer software product can be loaded into and run by the exemplary data processing system of FIG. 3.

Integrated Circuit Product. The sequences of instructions which when executed cause the method described herein to be performed by the exemplary data processing system of FIG. 3 can be contained in an integrated circuit product including a coprocessor or memory according to one embodiment of the invention. This integrated circuit product can be installed in the exemplary data processing system of FIG. 3.

Although preferred embodiments of the invention have been described herein, it will be understood by those skilled in the art that variations may be made thereto without departing from the spirit of the invention or the scope of the appended claims.